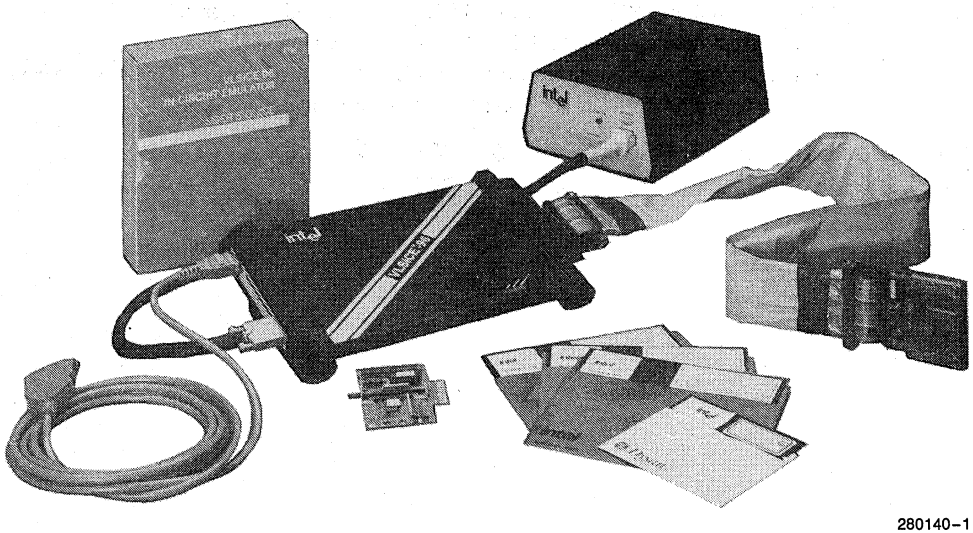
VLSiCE™-96P
IN-CIRCUIT EMULATOR FOR THE MCS®-96 FAMILY OF MICROCONTROLLERS

- **Precise Real-Time Emulation of the MCS®-96 Family of Components**
- **64K of Mappable Memory for Early Software Debug and (EP)ROM Simulation**
- **A 4K-Entry Trace Buffer for Storing a Real-Time Execution History, Including Both Code and Data Flows**
- **Multistate Break and Trace Qualified on Execution Addresses, Data Addresses, and Values (Both External and Internal RAM), Opcodes and Selected PSW Flags**
- **Fast Break and Dynamic Trace**
- **Shadow Registers Allow Reading Many 8096 Write-Only and Writing Many Read-Only Registers**
- **Symbolic Debugging Allows Accesses to Memory Locations and Program Variables (Including Dynamic Variables) Using Program-Defined Names**
- **Equipped with the Integrated Command Directory (ICD) Which Provides**
  - **— An On-Line Help File**
  - **— A Dynamic Syntax Menu**
  - **— Dynamic Command-Entry Error Checking**
- **Serially Linked to Intel Series III/IV Hosts or IBM* PC-XT and AT**

The VLSiCE™-96P In-Circuit Emulator is a debugging and test tool that is used for development of the hardware and software of a prototype system based on the MCS®-96 family of microcontrollers.



280140-1

*IBM is a trademark of International Business Machines.

November 1986
Order Number: 280140-003

INTRODUCTION

The VLSiCE-96P emulator allows hardware and software development of a design project to proceed simultaneously. With the VLSiCE-96P emulator, prototype hardware can be added to the system as it is designed and software can be developed prior to the completion of the hardware prototype. Software and hardware integration then occurs while the product is being developed.

The VLSiCE-96P emulator assists four stages of development:

- Software debugging
- Hardware development
- System integration
- System test

Software Debugging

The VLSiCE-96P emulator can be operated without being connected to the user's prototype or before any of the user's hardware is available. In this stand-alone mode, the VLSiCE-96P emulator can be used to facilitate program development.

Hardware Debugging

The VLSiCE-96P emulator's precise characteristics that match the controller and full-speed operation make it a valuable tool for debugging hardware, including time-critical serial port and timer interfaces.

System Integration

Integration of software and hardware can begin when the microcontroller socket is connected to any functional element of the user system hardware. As each section of the user's hardware is completed, it can be added to the prototype. Thus, each section of the hardware and software can be system tested in real-time operation as it becomes available.

System Test

When the prototype is complete, it is tested with the final version of the system software. The VLSiCE-96P emulator is then used for real-time emulation of the microcontroller to debug the system as a completed unit.

The final product verification test may be performed using the ROM or EPROM version of the microcontroller. Thus, the VLSiCE-96P emulator provides the ability to debug a prototype or production system at any stage in its development without introducing extraneous hardware or software test tools.

PHYSICAL DESCRIPTION

The VLSiCE-96P emulator consists of the following components (see Figure 1):

- Power supply
- AC and DC power cables
- Serial cable (host-specific)
- Controller pod
- User cable assembly (consisting of the user cable and processor module)
- Crystal power accessory (CPA)
- Multi-synchronous accessory (MSA)
- 48-pin DIP adaptor
  68-pin PGA adaptor
  68-pin PLCC adaptor (optional)
- Software (includes the VLSiCE-96P emulator software, diagnostic software, and tutorial)

The power supply connects to the controller pod via the DC power cable. There are several voltage options available for the power supply determined by the settings of the switches on the back of the supply.

The controller pod contains 64K of ICE memory, a 4K-entry trace buffer, and circuitry which provides communication between the host and the emulator.

The serial cable connects the host system to the controller pod. The serial cable has electrical specifications similar to the RS-232C standard.

The processor module contains a special version of the Intel 8096 microcontroller, called the emulation processor. This chip performs real-time and single-step execution of a program's object code for debugging purposes and replaces the target system microcontroller.

The crystal power accessory (CPA) is a small detachable board that connects to the back of the controller pod and is used to run the VLSiCE-96P emulator in the stand-alone mode. It is also used when running customer confidence tests. In the stand-alone mode, the user plug on the user cable is connected through the 68-pin PGA adaptor to the CPA. The CPA supplies clock and power. Stand-alone mode is used to test and debug software prior to the availability of hardware.

The multi-synchronous accessory can be used to connect up to 21 multi-ICE compatible emulators to-

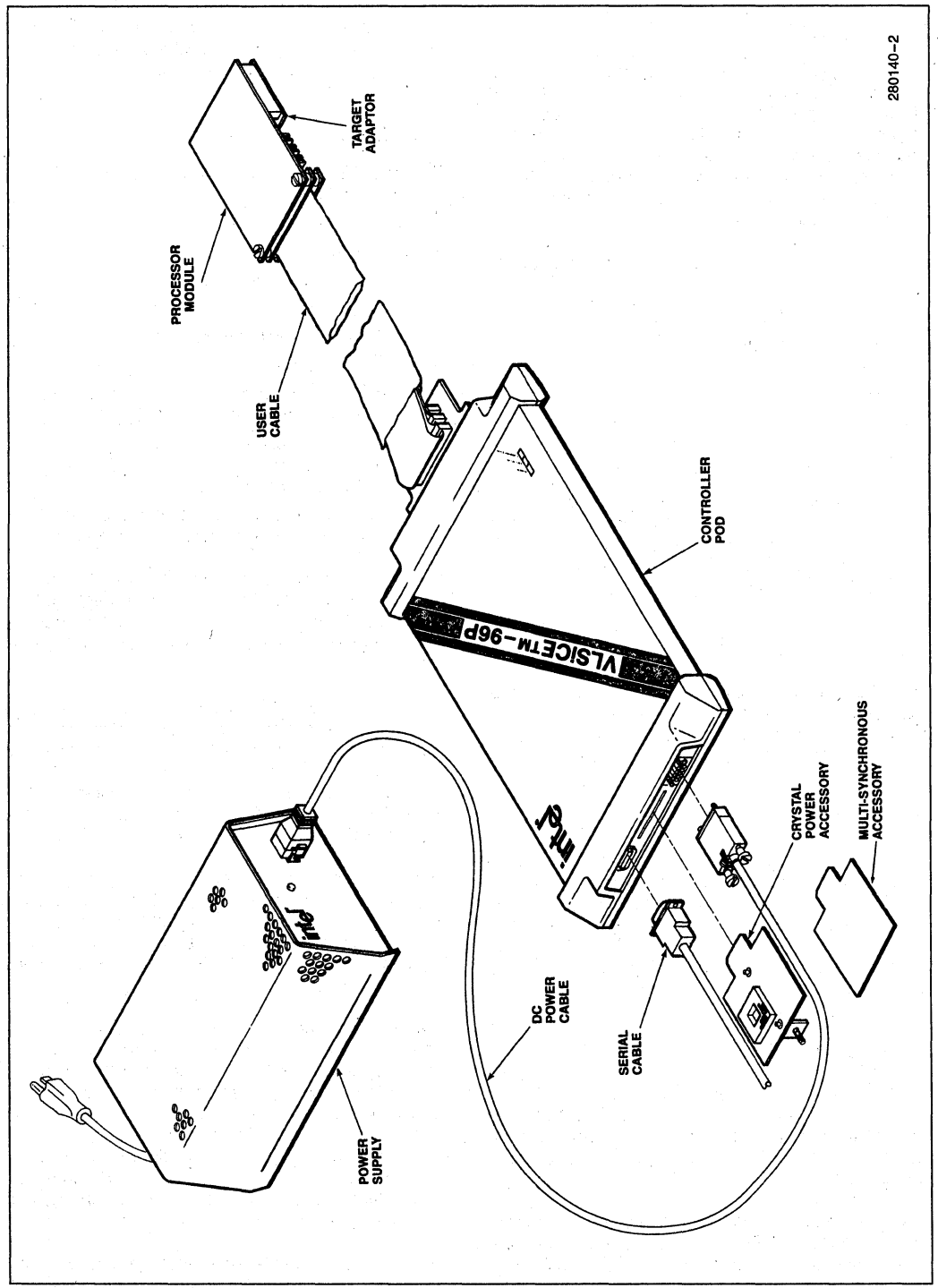


Figure 1. The VLSICE™-96P Emulator

gether for synchronous GO and BREAK emulation. It can also be used with other debug equipment such as logic analyzers and oscilloscopes.

Figure 1 shows a drawing of the VLSiCE-96P emulator.

VLSiCE software fully supports all mnemonics, object file formats, and symbolic references generated by Intel's ASM 96, PL/M 96, and C96.

The on-line tutorial is written in VLSiCE-96P command language. Thus, the user is able to interact with and use the VLSiCE-96P emulator while executing the tutorial.

A comprehensive set of documentation is included with the VLSiCE-96P emulator.

VLSiCE™-96P EMULATOR FEATURES

The VLSiCE-96P emulator has been created to assist a product designer in developing, debugging and testing designs incorporating the MCS-96 family of microcontrollers. The following are some of the VLSiCE-96P features:

Emulation

Emulation is the controlled execution of the prototype software in the prototype hardware or in an artificial hardware environment that duplicates the microcontroller of the prototype system. Emulation is a transparent process that happens in real-time. The execution of prototype software is facilitated through the VLSiCE-96P command language.

Memory Mapping

There is 64K of zero-waitstate, high-speed mappable memory available. This memory space can be mapped to either the user prototype system or to the on-board VLSiCE-96P memory space in 1K-byte blocks on 1K-byte boundaries. Mapping memory to the VLSiCE-96P system allows software development to proceed before prototype hardware is available. Memory mapping also gives the VLSiCE-96P emulator the capability to simulate the 8K-bytes of (EP)ROM on those versions of the chip for code verification and validation.

Memory Examination and Modification

The memory space for the MCS-96 component and its target hardware is accessible through the emulator. The VLSiCE-96P software allows the component's special function registers to be accessed mnemonically (e.g. AD RESULT, INT MASK). A significant benefit to the user is the ability of the VLSiCE-96P software to read many of the write-only registers (e.g. AD_COMMAND, PWM_CONTROL) and to write many of the read-only registers (e.g. AD_RESULT, SBUFRX). Data can be displayed or modified in several bases: hex, decimal, and binary, and in these formats: ASCII, real and integer. Program code can be disassembled and displayed with opcode mnemonics. It also can be modified with standard assembler statements.

Symbolic debugging is used to specify memory locations by their symbolic references. A symbolic reference is a procedure name, line number, or label in the user program that corresponds to a location. Using symbolics to reference program locations is a mnemonic way of accessing the program.

Some typical symbolic functions include:

- Changing or inspecting the value of a program variable by using its symbolic name, rather than the address of the memory location.
- Defining break and trace events using symbolic references.
- Referencing static variables, dynamic (stack-resident) variables, based variables, and record structures combining primitive data types. The primitive data types are ADDRESS, BOOLEAN, BYTE, CHAR (character), WORD, DWORD (double word), INTEGER, LONGINT, SHORTINT, and REAL.

The VLSiCE-96P system maintains a virtual symbol table for program symbols making it possible for the table to exist without fitting entirely into host RAM memory. The size of the virtual symbol table is constrained only by the capacity of the disk.

Breakpoint Specifications

Breakpoints allow halting of a user program in order to examine the effect of the program's execution on the user prototype system. Breakpoints can be defined as execution addresses, data addresses and data values (both external and **internal RAM**, opcodes, and selected bids of the PSW flag. These breaks can also be arranged to occur over a range of addresses. After a break the user program can resume execution from where it left off.

Trace Specifications

Tracing can be triggered with the same conditions set for breaking. The trace buffer is displayed as disassembled instructions, data fetches and stores, and with the timetag showing the relative time at

```
hlt> PRINT NEWEST 8 CYCLES
FRAME ADDRESS CODE          MNEMONIC OPERANDS      TIME

(1020) 300A C82A            PUSH 2A                172 µs
       [002AH]=0087H(R) [0018H]=0028H(R)     [0018H]=0026H(W)
       [0026H]=0087H(W)

(1021) 300C 64322A          ADD 2A,32              175 µs
       [0032H]=0010H(R) [002AH]=0087H(R)     [002AH=0097H(W)

(1022) 300F 6975002A        SUB 2A,#75             180 µs
       [002AH]=0097H(R) [002AH]=0022H(W)
.
.
.
.
.
(1027) 3010 CC2A0002        POP 2A                 205 µs
       [0018H]=0026H(R) [0026H]=0087H(R)     [0018H=0028H(W)
       [002AH]=0087H(W)
```

Figure 2. The Trace Buffer Display

which the program executed each instruction. Figure 2 shows a trace display as a result of the PRINT command.

Normally, the VLSiCE-96P system traces program activity while the user program executes. With a trace specification, tracing can be specified to occur only when specific conditions are met during execution. The trace is collected in a buffer that can collect data on up to 4K entries of information during emulation.

The trace buffer can be examined during halt mode or if non-stop emulation is desired, the trace can be examined while emulation continues. If this second option is selected, trace collection stops briefly while the trace buffer is uploaded to the host.

Arming and Triggering

The VLSiCE-96P command language allows specification of complex events with up to 8 states, each with several conditions. For example, a specification can be made that causes a break to occur when a variable is written only within a certain procedure. The execution of the procedure is the arm condition; the variable modification is the break condition. The arm condition is an optional part of a break/trace sequence in the VLSiCE emulator. A set of arm conditions can be used to ensure that a system break is not possible until all required qualifying conditions are satisfied.

Procedures

Debugging procedures (PROCS) are a user-named group of VLSiCE commands that are executed sequentially. Procs can simulate missing hardware or software, collect debug information, and make troubleshooting decisions. They can be copied to text files on disk, then included from the file into the command sequence in later test sessions.

Procedures can also serve as programmable diagnostics, implementing new emulator commands for special purposes or to increase generality.

Dynamic Tracing

The trace buffer can be dynamically accessed during emulation. Any form of the PRINT command can be entered and the specified portion of the trace buffer is displayed. This allows real-time display of processor activity. However, displaying the trace buffer during emulation stops collection of trace and some trace information could be lost.

On-Line Syntax Guide

A special syntax guide called the Integrated Command directory (ICD), at the bottom of the display screen, aids in creating syntactically correct command lines. Figure 3 shows an example of the ICD for the GO command.

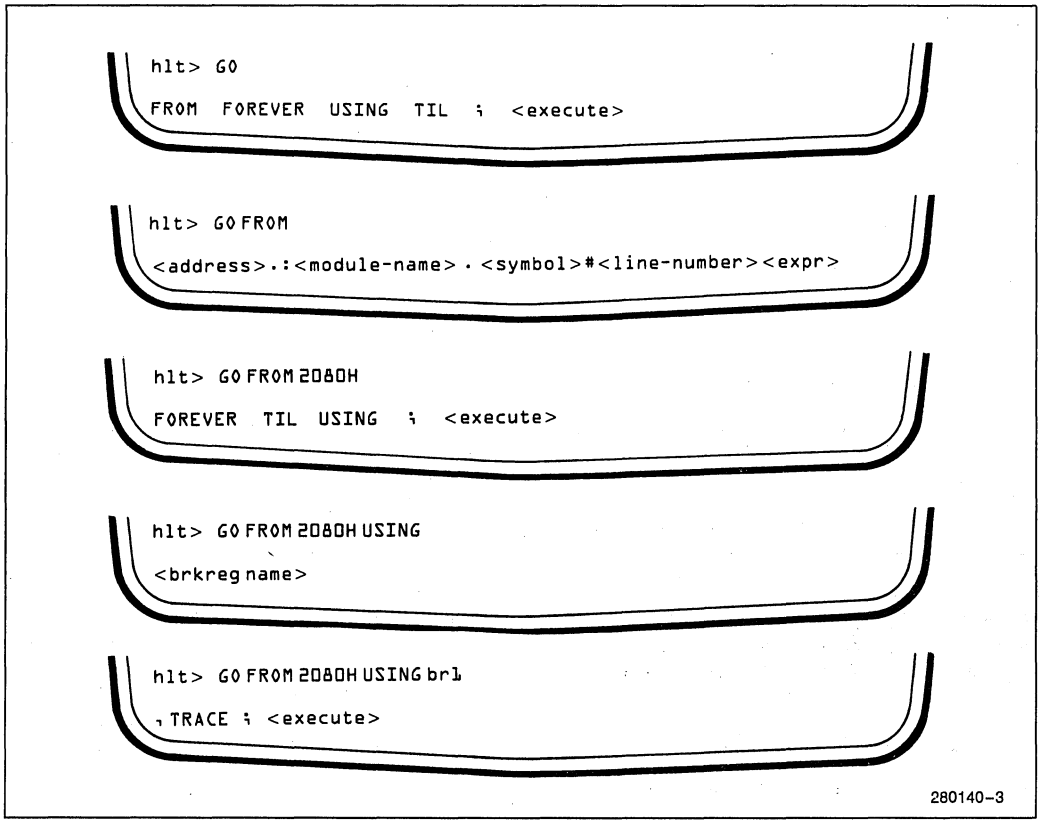


Figure 3. The Integrated Command Directory for the GO Command

HELP

This feature provides assistance with the emulator commands through the host system terminal. HELP is available for most of the commands. Figure 4 shows one of the commands HELP can be obtained for.

DESIGN CONSIDERATIONS

There are design considerations that the user should be aware of before designing with the VLSiCE-96P emulator.

Electrical Considerations

The user pin timings and loadings are identical to the 8096 component except as noted below. Also, the RESET and CLKOUT pins have an additional loading of 1 μA and 10 pF. The Non-Maskable Interrupt (NMI) is not supported.

	Min.	Max.
Clock Frequency	6 MHz	10 MHz
V_{CC}	4.75V	5.25V
I_{CC}(1)		400 mA

NOTE:
1. All outputs disconnected.

Mechanical Considerations

The user plug is at the end of a three foot flexible cable. Adequate spacing must be provided on the target system to allow the processor module to be inserted into the target system.

The height of the processor module and target adaptor may pose a problem for multiple board system prototypes that need to be debugged and tested. Be sure that the space between the boards is greater than 1½" to allow for the processor module and target adaptor.

```
hlt> HELP ASM
The ASM command displays or modifies memory as 8096 assembler mnemonics.
The syntax is:

 ASM<partition>    ::=<partition> [='<asm96-inst>'    [,'<asm-96inst>'*]]
                   <address> <cr>

where:

<partition>specifies the area of memory to be displayed and modified.

<asm96-inst>specifies the 8096 assembly instruction to be assembled.

<address> is any valid 8096 address.

<cr> indicates a carriage return.

The "ASM <address>=" syntax puts the user in line-mode, displaying the
current address at which the instruction will be placed and not requiring the
quotes around the instructions.

Please see the VLSiCE-96P Reference Manual for more detailed information.
```

Figure 4. HELP Screen

Figure 5 shows the dimensions for the processor module and target adaptor. In the figure, please note the location of pin 1 on each target adaptor.

Limitations and Restrictions

- The non-maskable interrupt (NMI) is not supported.
- High Speed Input/Output (HSIO) is emulated, but can not be read or written to during interrogation mode.
- If a break occurs immediately before the JVT instruction, the VT flag is cleared. There is no way to then tell if the flag was set before re-entering emulation.

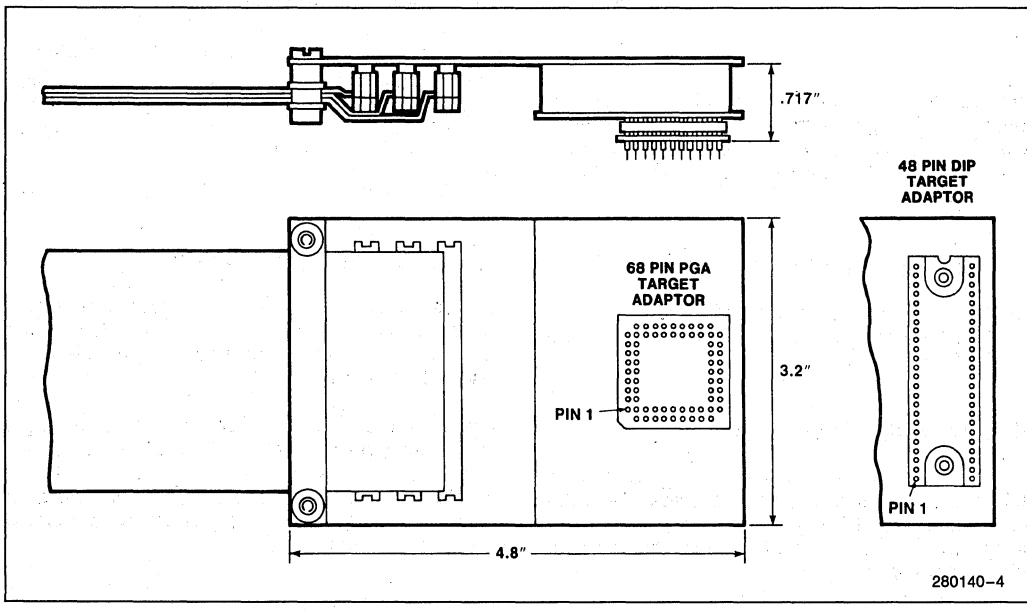


Figure 5. Dimensions for the Processor Module and Target Adaptor

- If a read access of SP_STAT or IOS1 occurs at the same time that an interrupt sets a bit in the accessed register, there is a possibility that the new interrupt flag will be lost, because read accesses to the SP_STAT or IOS1 registers clear these registers. A software work-around for this problem is provided.
- Bit 7 of Port 0 is inverted when the digital value of the port is read.
- The counters for the pulse width modulator (PWM) and hardware Timer 1 can be out of sync if either are disabled during interrogation. They can be re-synced with a user reset of the system.
- The READY pin is not supported. It should be tied high (+5V).
- The system presently operates up to 10 MHz.
- Ports 3 and 4 cannot be read by code during emulation or by the host during interrogation when they are used as ports (EA/=1).

The VLSiCE-96P system has some limitations that are inherent in the 809X-90 core of the emulation processor. These limitations are:

- The displacement portion of an indexed, three word multiply may not be in the range of 200H through 17FFH inclusive.
- The EXT instruction never sets the N flag, and always sets the Z flag. The EXTB instruction works correctly.
- The zero flag is either set or cleared by the Add or Subtract with carry instructions as appropriate.
- Trying to read, modify, or write the interrupt pending register may cause interrupts to be missed during execution of the instruction.
- The V and VT flags may indicate an overflow after a signed divide when no overflow has occurred.
- If an event on an HSI pin set to look for every eighth transition occurs less than 16 state times after an event on any other pin, then the divide by 8 event is recorded twice in the HSI FIFO. The time tag of the duplicate FIFO entry is equal to that of the initial entry plus one.
- An event occurring within 16 state times of a prior event on the same HSI line may not be recorded. Additionally, an event occurring within 16 state times of a prior event on another HSI line may be recorded with a time tag one count earlier than expected. Events are defined as the condition the line is set to trigger on.
- Software timer interrupts cannot be generated by the HSO commands that reset Timer 2 or start an A to D conversion.
- The serial port is not tested in mode 0. The receive function in this mode does not work correctly.
- Loading the baud rate register with 8000H (maximum baud rate, internal clock) may cause an 11 millisecond delay (at Fosc = 12 MHz) before the port is properly initialized. After initialization the port works properly.
- To be used as outputs, ports 3 and 4 each must be addressed as words but written to as bytes. To write to Port 3 use 'ST temp, 1ffeh', where the low byte of 'temp' contains the data for the port. To write to Port 4, use the DCB operator to generate the opcode sequence 0C3H, 001H, 0FFH, 01FH, (temp), where the high byte of temp contains the data for the port. Ports 3 and 4 do not work as inputs.
- The watchdog timer does not run after a chip reset until a 01EH followed by a 0E1H is written to the watchdog timer register. When this is done, the watchdog timer functions as described in the 8096 Users Manual until the next chip reset. This feature permits disabling of the watchdog by not writing to it.
- External interrupts on P0.7 are sampled every state time instead of every eight state times.
- The baud rate generated in the external clock can be computed using this formula:

  Baud Rate = Input Frequency/(16*B)

This formula does not work in mode 0 and assumes T2CLK is the source of the input frequency.

- When more than one HSO event occurs with interrupts occurring at the same time, multiple HSO interrupts may occur. This is because HSO interrupts are internal events and are not synchronized to Timer 1.
- Locations 2012H through 207FH in external memory must be filled with the hex value 0FFFFH to ensure compatibility with future parts. The internal locations in this range are still reserved for the factory test code.
- Neither the source nor the destination addresses of the Multiply or Divide instructions can be a writable special function register.
- The special function registers may not be used as base or index registers for indexed or indirect instructions.
- Several of the special function registers can only be accessed as words, while others only as bytes. These restrictions are listed in the 8096 Users Manual.

SPECIFICATIONS

Host Requirements

Disk drives— Dual floppy or 1 hard disk and 1 floppy drive required (10M-byte Winchester optional).

An IBM PC XT or PC AT with 512K RAM and hard disk. Intel recommends PC-DOS 3.0 or later. Earlier versions of PC-DOS may be acceptable.

Other peripherals as desired.

VLSiCE-96P Software Package

VLSiCE-96P emulator software
VLSiCE-96P confidence tests
VLSiCE-96P tutorial software

System Performance

Mappable zero wait-state memory (zero wait-states up to 12 MHz)	Min. 0K-bytes Max. 64K-bytes	Mappable to user or ICE memory in 1K blocks on 1K boundaries.

Trace buffer 4K × 48 bits

Virtual symbol table— A maximum or 61K-bytes of host memory space is available for the virtual symbol table (VST). The rest of the VST resides on disk and is paged in and out as needed.

Physical Characteristics

Controller Pod

Width: 8¼" (21 cm)
Height: 1½" (3.8 cm)
Depth: 13½" (34.3 cm)
Weight: 4 lbs (1.85 kg)

Power Supply

Width: 7⅝" (18.1 cm)
Height: 4" (10.06 cm)
Depth: 11" (27.97 cm)
Weight: 15 lbs (6.1 kg)

User Cable

3' (.944m)

Serial Cable

12' (3.6m)

Electrical Characteristics

Power Supply

100–120V or 220–240V (selectable)
50–60 Hz
2 amps (AC max)@ 120V
1 amp (AC max)@ 240V

Environmental Characteristics

Operating temperature: 0°C to 40°C (32°C to 104°F)
Operating humidity: Maximum of 85% relative humidity, non-condensing

DOCUMENTATION

VLSiCE-96P In-Circuit Emulator User's Guide, order number 165814.

VLSiCE-96P In-Circuit Emulator Installation Supplement for Intel Hosts, order number 166477.

VLSiCE-96P In-Circuit Emulator Installation Supplement for IBM Hosts, order number 166478.

ORDERING INFORMATION

Base Hardware and Software

Order Code	Description
V096S96A	VLSiCE-96P emulation base and software on 8″ single density media for hosting on an Intel Series III. [Requires software license].
V096S96B	VLSiCE-96P emulation base and software on 8″ double density media for hosting on an Intel Series III. [Requires software license].
V096S96C	VLSiCE-96P emulation base and software on 5¼″ media for hosting on an Intel Series IV. [Requires software license].
V096S96D	VLSiCE-96P emulation base and software on 5¼″ media for hosting on an IBM PC-AT under PC-DOS 3.0. [Requires software license].

Serial Cables (must be ordered for complete system)

Order Code	Description
SCOM1	Serial communications cable is a 25-pin male to 9-pin male for hosting VLSiCE-96P on Intel Series III/IV.
SCOM2	Serial communications cable is a 25-pin female to 9-pin male for hosting VLSiCE-96P on IBM PC-XT.
SCOM3	Serial communications cable is a 9-pin female to 9-pin male for hosting VLSiCE-96P on IBM PC-AT.
TA096E	68 pin PLCC adaptor board.

Software Only

Order Code	Description
S096AP	Software for host, probe, diagnostic and tutorial on 8″ single sided media for use with a Series III. [Requires software license].
S096BP	Software for host, probe, diagnostic and tutorial on 8″ double sided media for use with a Series III. [Requires software license].
S096CP	Software for host, probe, diagnostic and tutorial on 5¼″ media for use with a Series IV. [Requires software license].
S096DP	Software for host, probe, diagnostic and tutorial on 5¼″ media for use with a IBM PC AT or PC XT (requires PC DOS 3.0 or later). [Requires software license].

Other Useful Intel MCS®-96 Debug and Development Support Products

Order Code	Description
I86ASM96	Consists of the ASM 96 macro assembler that translates symbolic assembly language mnemonics into relocatable object code, and the RL96 linker and relocator program that links modules generated by ASM 96 and PL/M 96 and locates the linked object modules to absolute memory locations. System requirements and Intellec® System running iNDX.
I86PLM96	Consists of the PL/M 96 compiler that provides high level programming language support, the LIB 96 utility that creates and maintains libraries of software object modules, the FPAL96 floating point arithmetic library, and the RL96 linker and relocator program that links modules generated by ASM 96 and PL/M 96 and locates the linked object modules to absolute memory locations. System requirements and Intellec® System running iNDX.
D86ASM96NL	ASM/R&L 96 for PC-DOS. It contains a macro assembler, a linker/locator utility, a floating point utility and a librarian. System requirements are an IBM PC AT or PC XT with 512 K of RAM and PC-DOS 3.0 or greater.

Order Number	Description
D86PLM96NL	PL/M 96 and R&L for PC-DOS. It contains a compiler, a linker/locator utility, a floating point utility and a librarian. System requirements are an IBM PC AT or PC XT with 512K of RAM and PC-DOS 3.0 or greater.
D86C96NL	C96 and R&L for PC-DOS. Contains a compiler linker/locator utility, and all standard C libraries, including STDIO. System requirements are an IBM PC AT or PC XT with 512K of RAM and PC-DOS 3.0 or greater.
SBE96SKIT	iSBE-96 single board emulator for use with the Series III/IV development systems. The kit contains: iSBE-96 Single Board Emulator MCS®-96 software support package for the Series III/IV development systems. iSBE-96 Series III/IV upgrade kit (cables and software needed to run on Intel Hosts).
SBE96DKIT	iSBE-96 single board emulator for use with the IBM PC AT and PC XT computer systems. The kit contains: iSBE-96 single board emulator MCS®-96 software support package for PC-DOS. iSBE-96 DOS upgrade kit (cables and software needed to run on the IBM PC AT or PC XT).

Running the iSBE-96 emulator on the Series II and iPDS system requires software from:

U.S. Software Corporation
5470 N.W. Innisbrook
Portland, OR 97229
Phone: 503-645-5043
International Telex: 4993875